CSE702 Week 3: "Objective" Metrics and Elo Ratings

"Objective" means raw counting, with no dependence on model training. With respect to a strong computer chess engine and span of search used as the benchmark, here are some core metrics:

- **T1**-Match (called **MMP** for Move-Match % by me): The % of playing the move listed first by the engine.
- **EV**-Match: includes a move of **e**qual **v**alue to the first move as a match. (Recommended in this paper, which called it CV for "coincidence value.") (Only for the first 5 listed moves.)
- **ACPL**: "Average Centipawn Loss"---means without scaling.
- **ASD**: Average Scaled Difference---see "When Data Serves Turkey" versus ACPL.
- **Err025**: Count of errors of 0.25 or more (not scaled).
- **T3-Match**: credits any of the first 3 listed moves by the engine.
- **T3thr50**: credits playing a top-3 move only if at most 0.50 inferior (not scaled).

It is widely opined that "smart cheaters" often play 2nd-best or 3rd-best moves to throw off detection via T1 or EV, provided the move is not too bad. Quite apart from cheating, IMHO T3thr50 is the most reflective of chess skill overall.

Our question is: **Should these quantities be strictly linear in the chess ratings of the players, all across the scale from neophyte to champion?** (And up to computers---up to the limits of resolution from programs themselves being the benchmark.) This presupposes that the population of players is in a good "steady state" with regard to ratings. This may fail for several reasons:

1. The update rule $R' = R + K \cdot (perf. - proj.)$ may not "mix" fast enough.
2. The population is not static: people (such as myself) leave having withdrawn more points than given, especially at the high end---but there is even more turnover at the low end.
3. External events may derail the correspondence of rating to skill---war/isolation/pandemic.

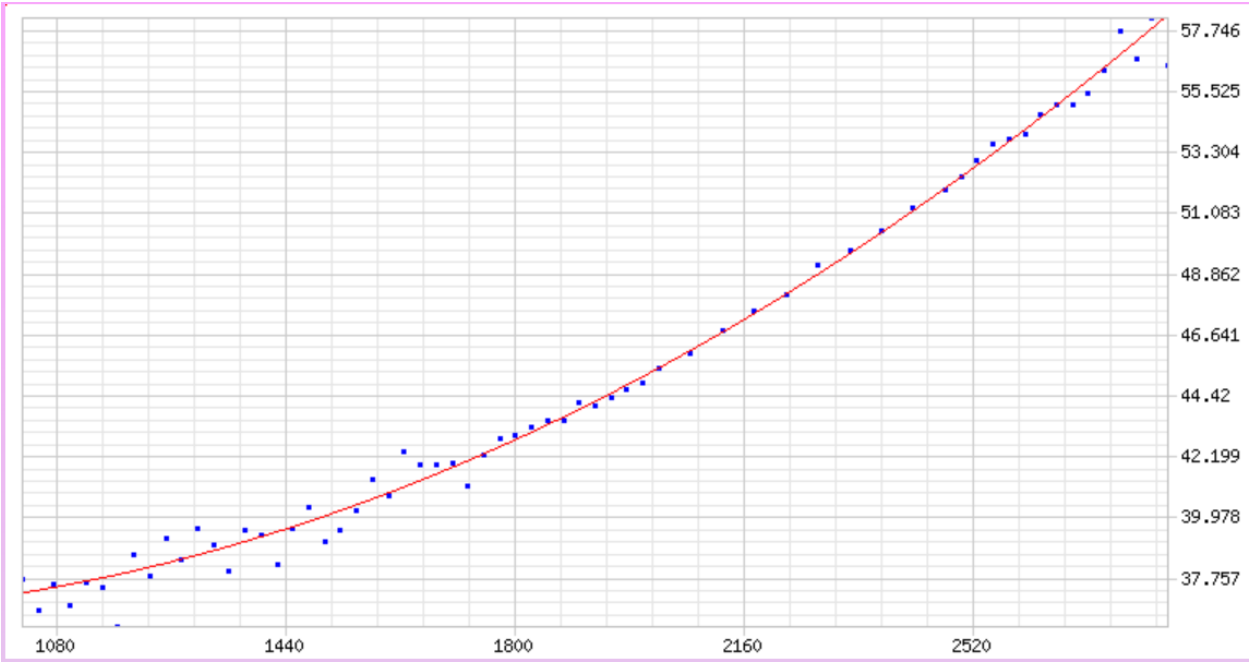
The fact of the system's absolute scale invariance heightens 3 as an issue: if everyone in a pocket of the world is 200 points underrated to the rest of the world, the pocket will still work *soundly* so long as it is isolated. Up thru 2011--2015, my work showed ratings at the high end---*at least where most recorded chess games are played, in Europe*---to have been remarkably stable since the late 1970s, nothing like the "150 Elo inflation" that was commonly alleged.

Reason 2, amplified by FIDE's policy of lowering the rating floor in the 2010s (and more recently changing how $proj.$ is computed when the rating difference is large), is Sonas's reason for ratings being out of whack below Elo 2000. He was aware of reason 3 but did not try correcting for "pandemic lag" as I do. I have reliable snapshots only thru the end of 2019, so let's see the state then. The charts use Andrew Que's "Polynomial Regression Online" web app.

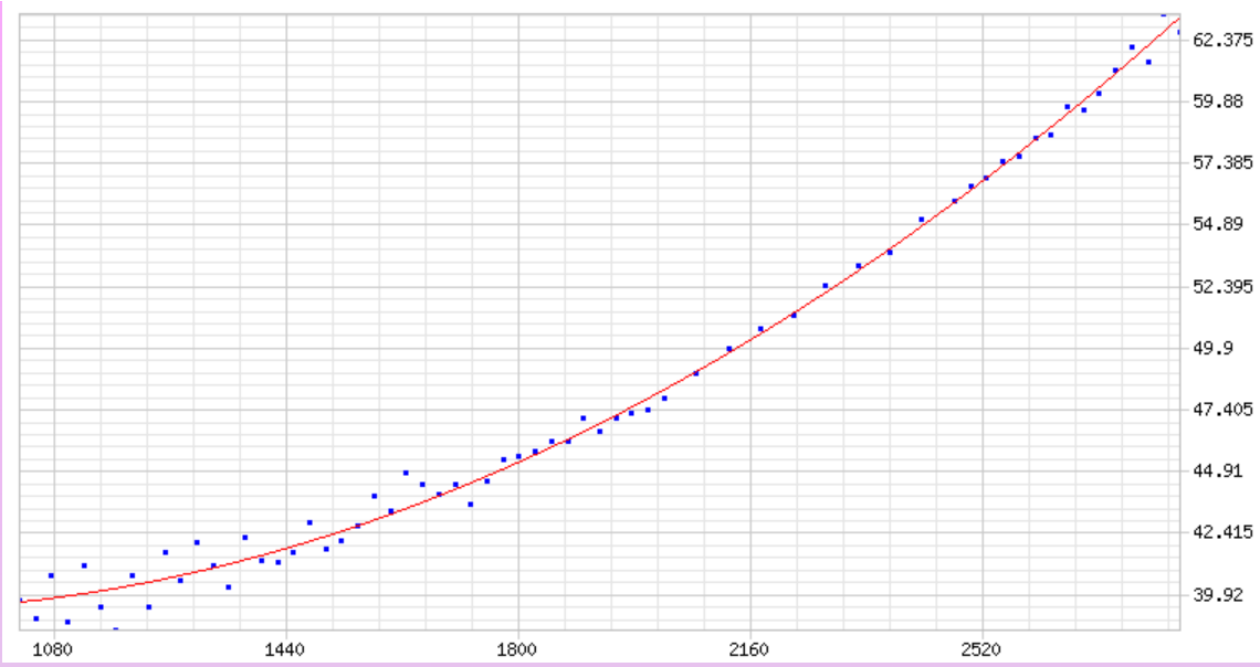
Graphs of These Metrics Versus Elo Ratings In 2010--2019

The data points for Elo 2050, 2100, 2150, 2200, 2250, 2300, 2350, 2400, and 2450 are missing on purpose. The samples for 2025 thru 2475 are already heftier than those for ratings above 2500 or below 2000. They have not yet been run with Stockfish 16 (in Multi-PV mode to variable depth 20--30).

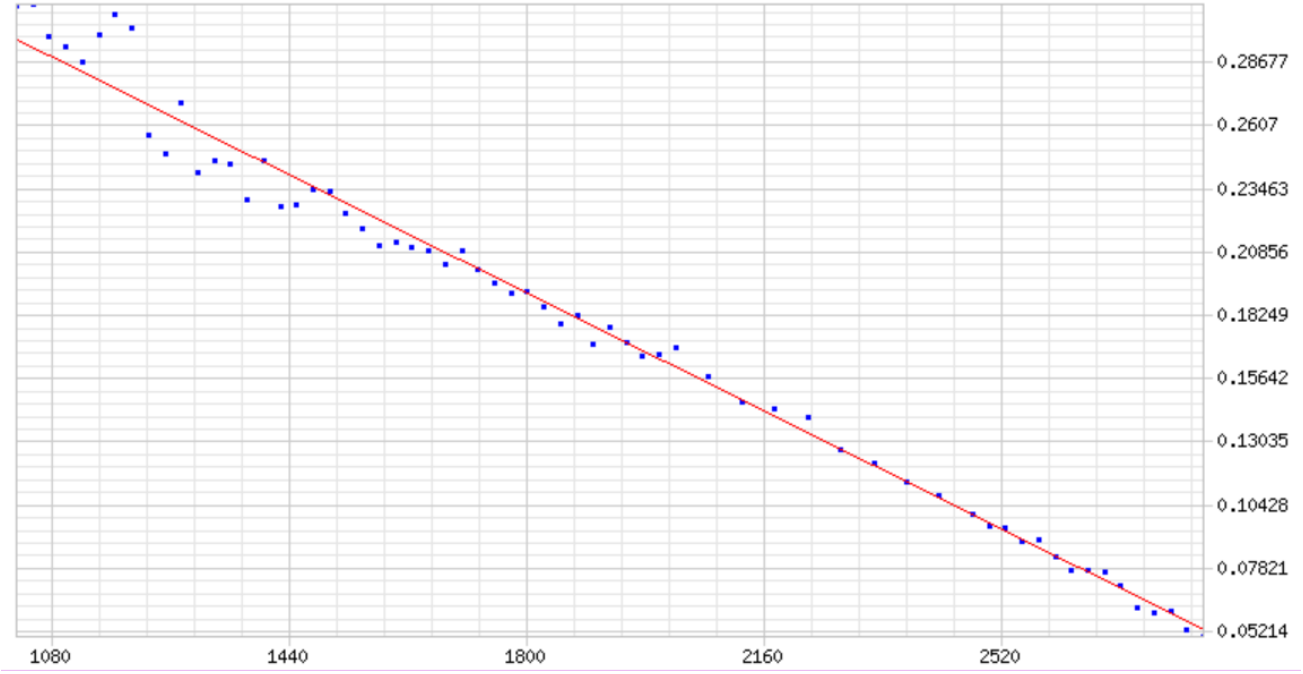
T1-Match:



EV-Match:

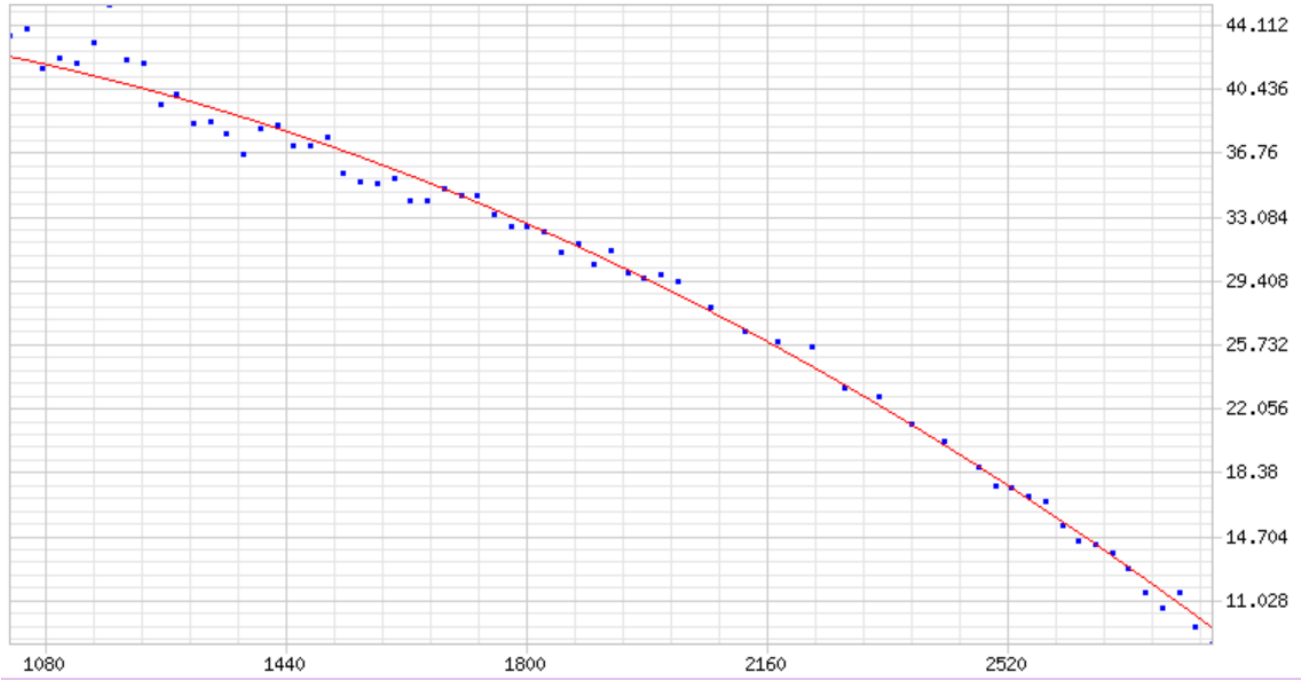


ASD: (See similar graphs with unscaled ACPL at the end.)



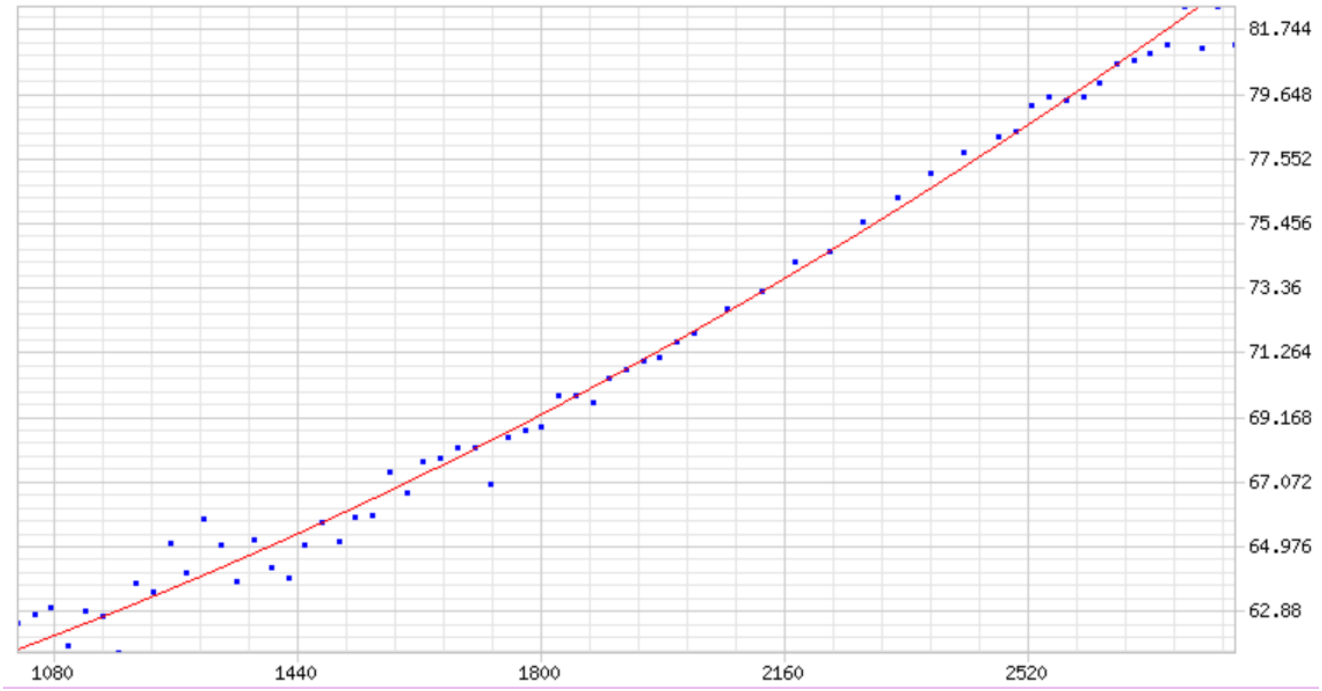
This still fits a straight line. The R^2 is 0.9861 weighted by the move sample size for each data point, 0.9863 unweighted. (This is a major reason I did not suspect the nonlinearity in T1 until 2018, well after the "Turkey" article.)

Error Count Err025:



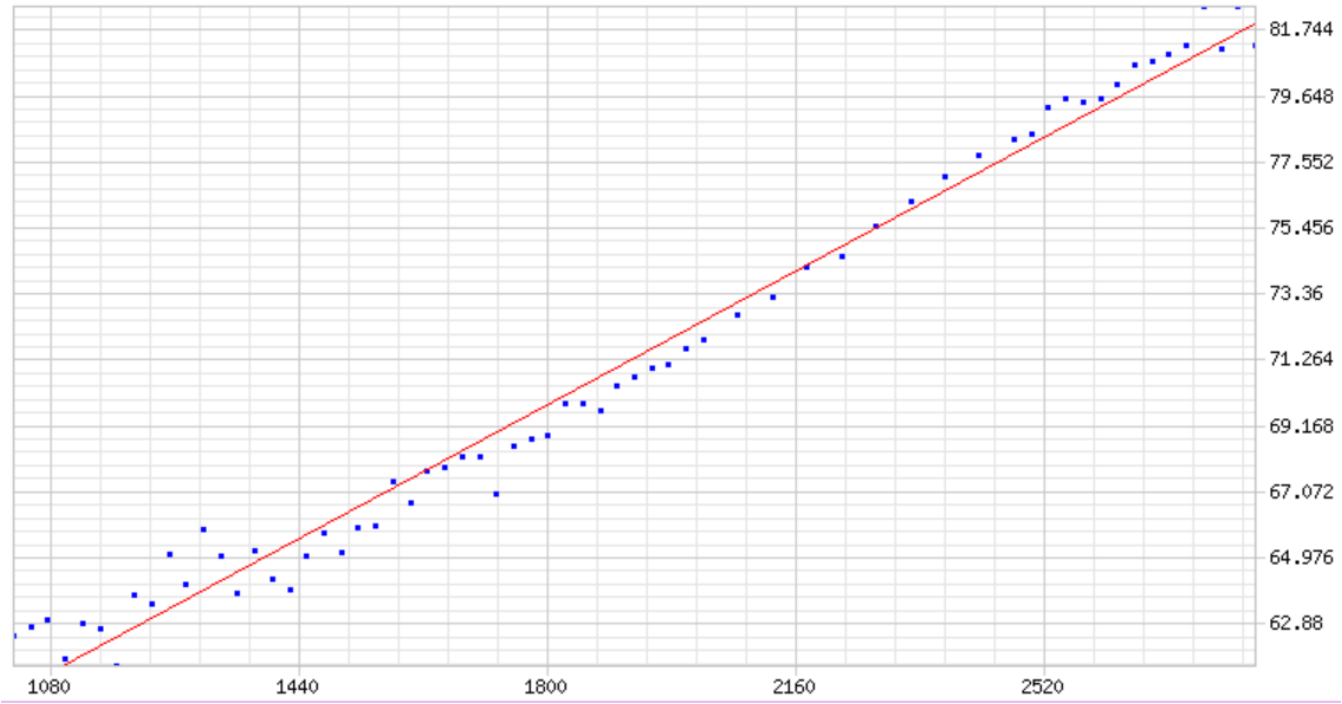
Again curved.

T3-Match:

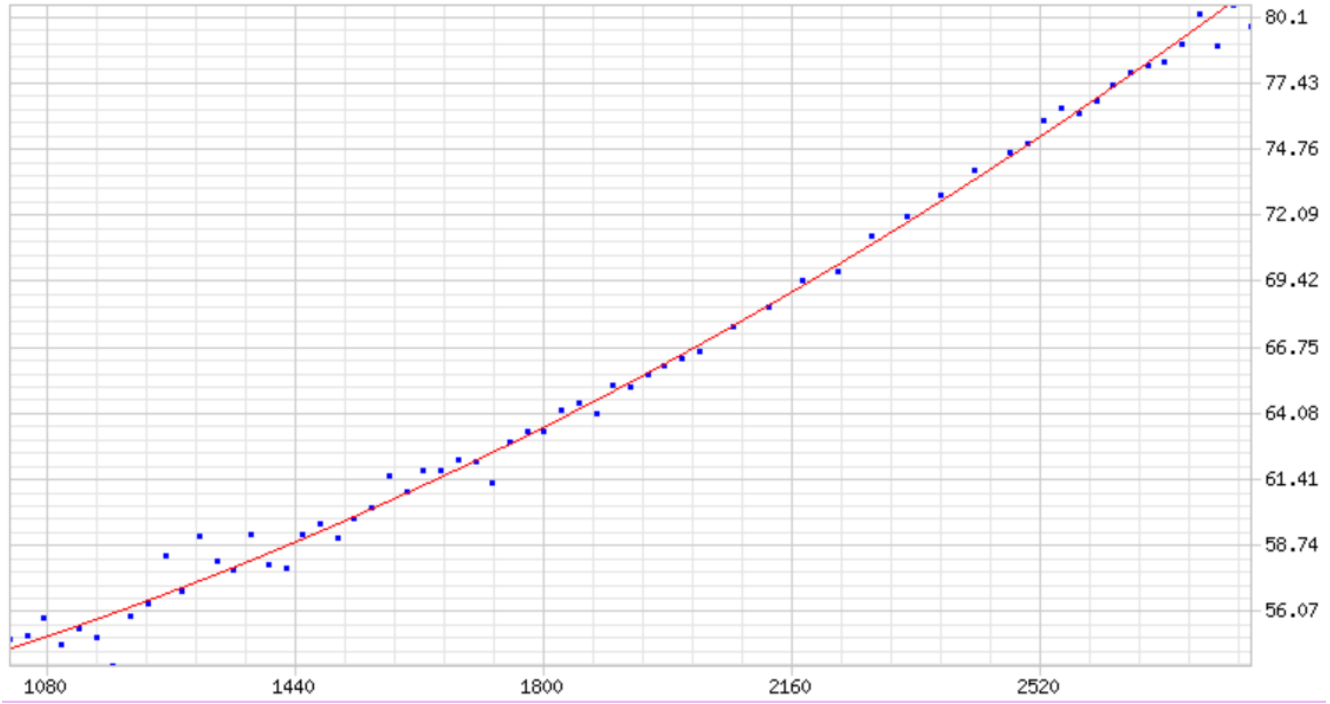


Still curved, with $R^2 = 0.9898...$

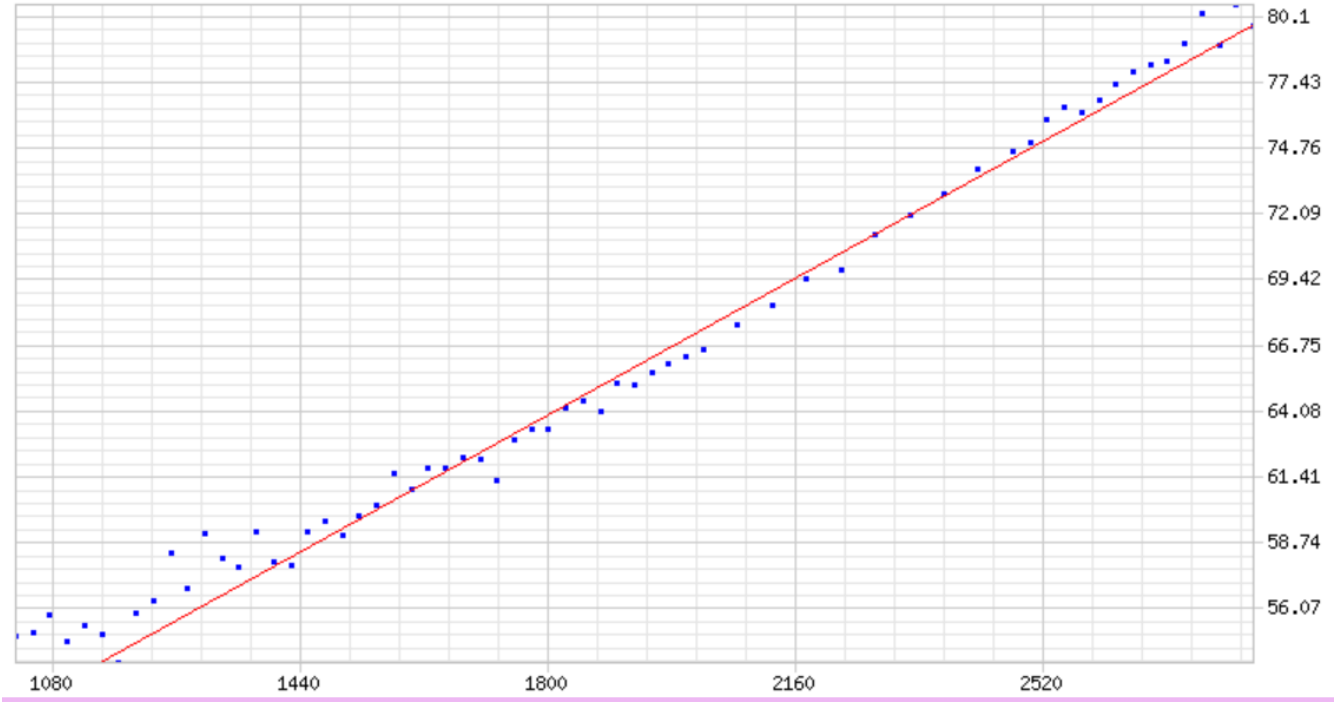
A line, however, is not bad either---$R^2 = 0.9787...$ weighted, $0.983...$ without:



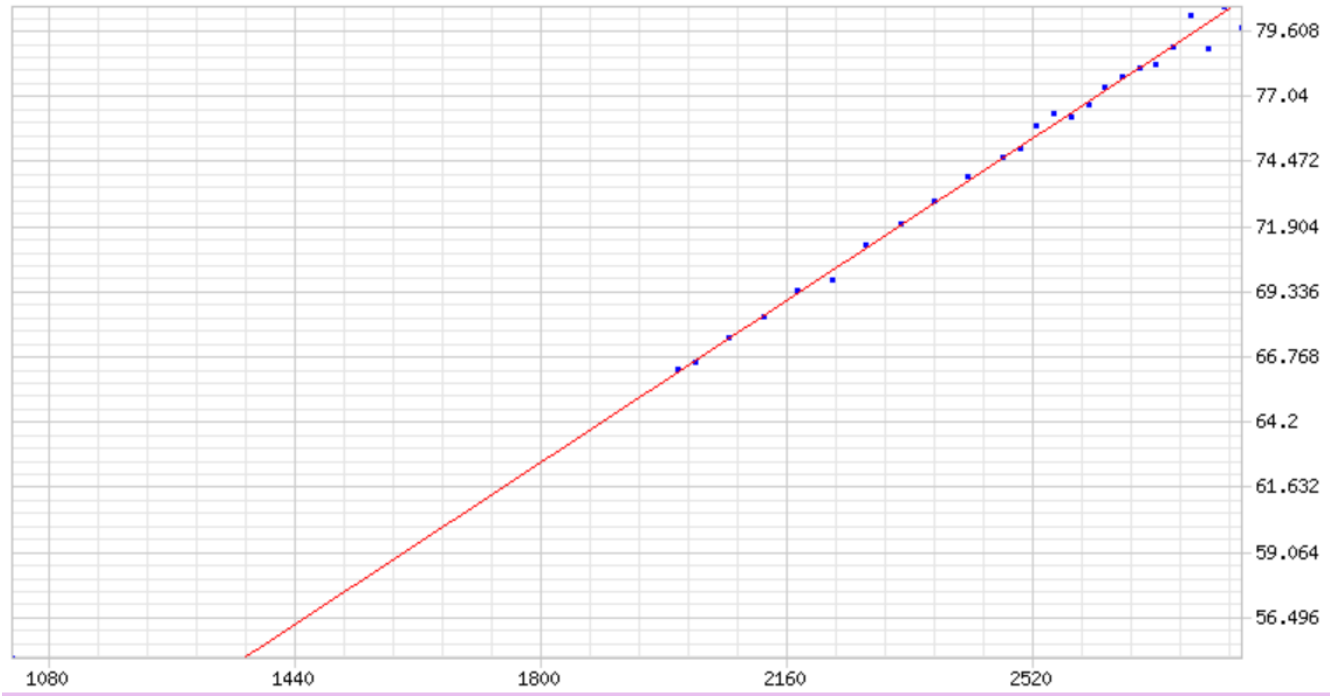
T3thr50:

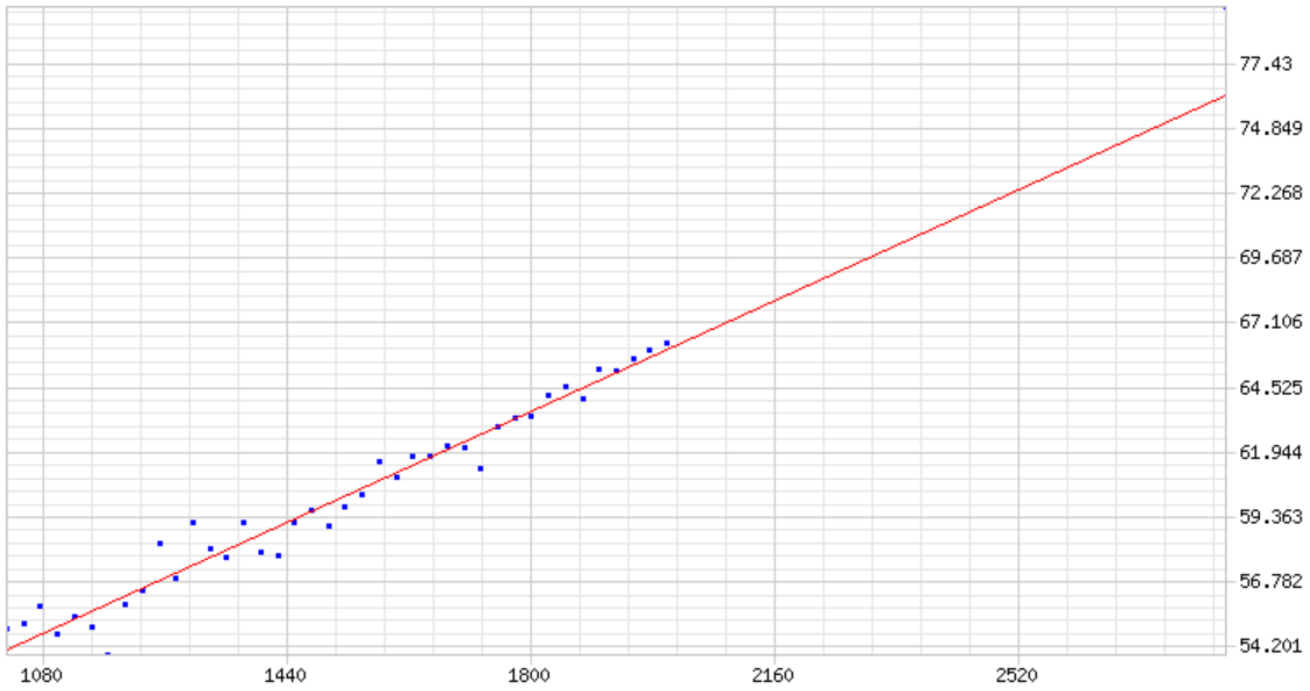


$R^2 = 0.994...$ for this. A line (with weighting turned on) looks less convincing:



More likely a kinked line: $R^2 = 0.9924...$ for the upper part, $R^2 = 0.9616...$ for the lower part.

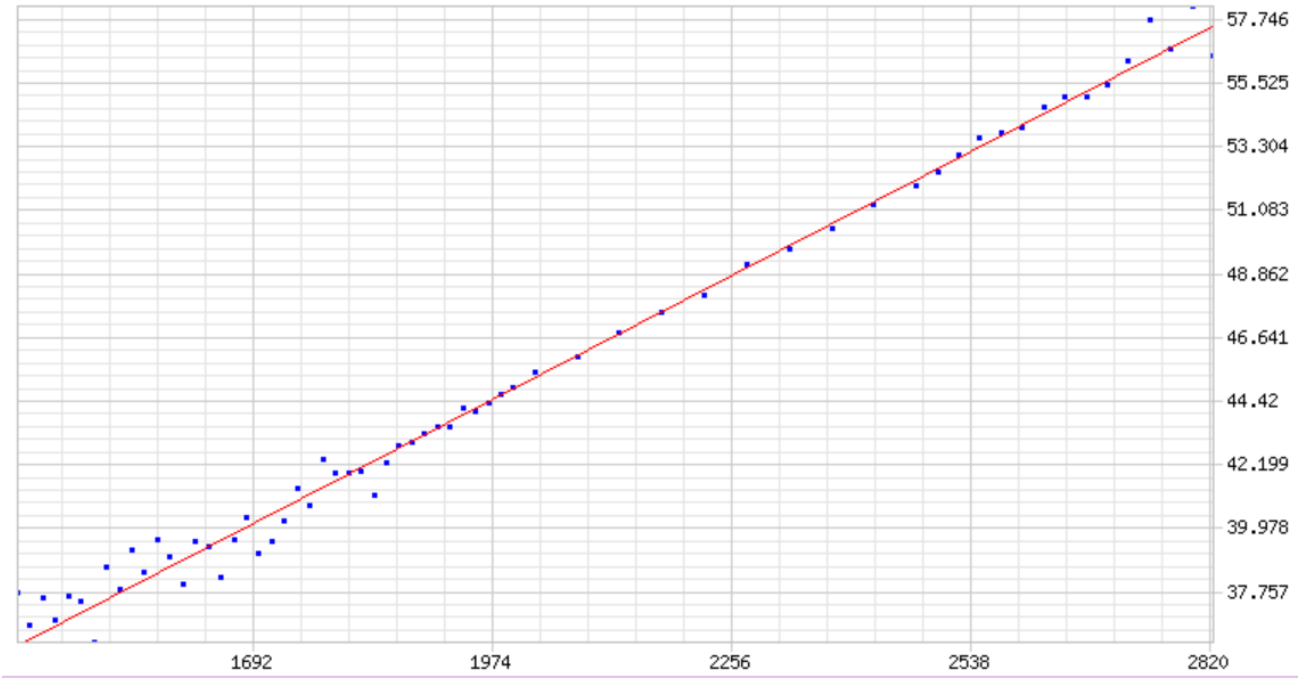




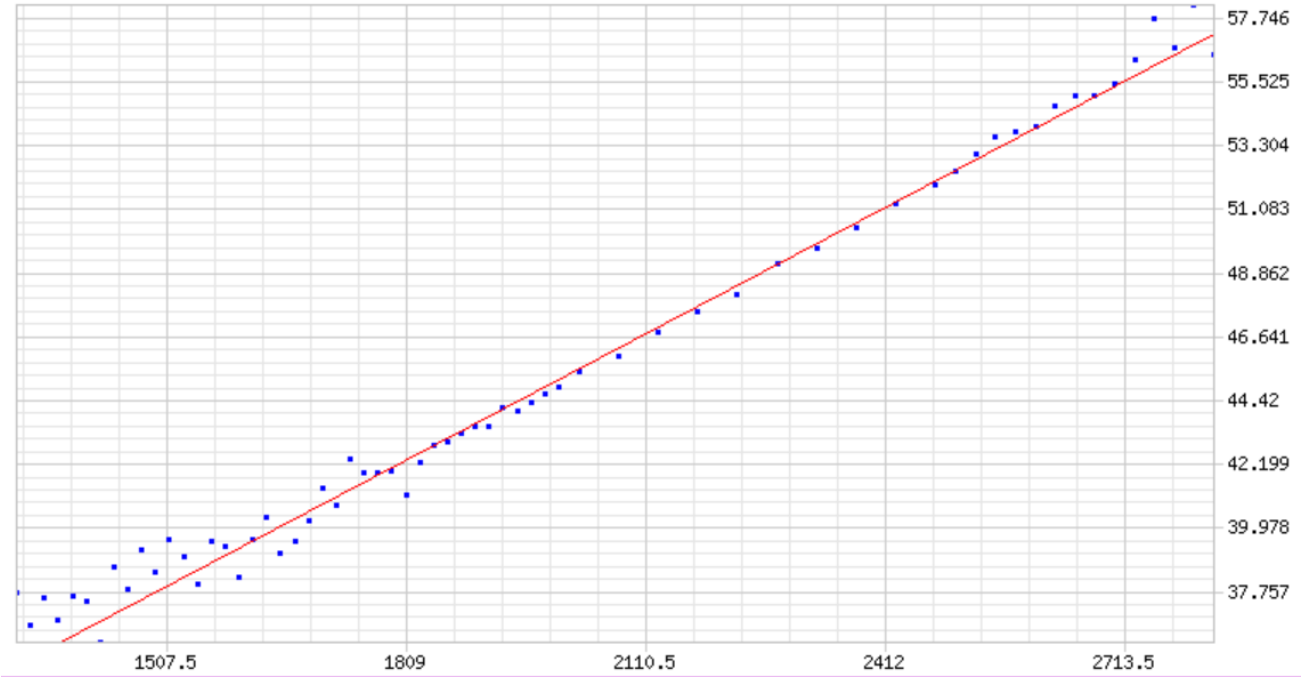
Graphs After the Sonas 400-Point "Compression"

This maps the interval [1000,2000] linearly onto the interval [1400,2000]. For example, a player rated 1500 is given 200 points more to make 1700, and a player rated 1250 goes up 300 to become 1550. A less-drastic change would map onto [1300,2000] instead.

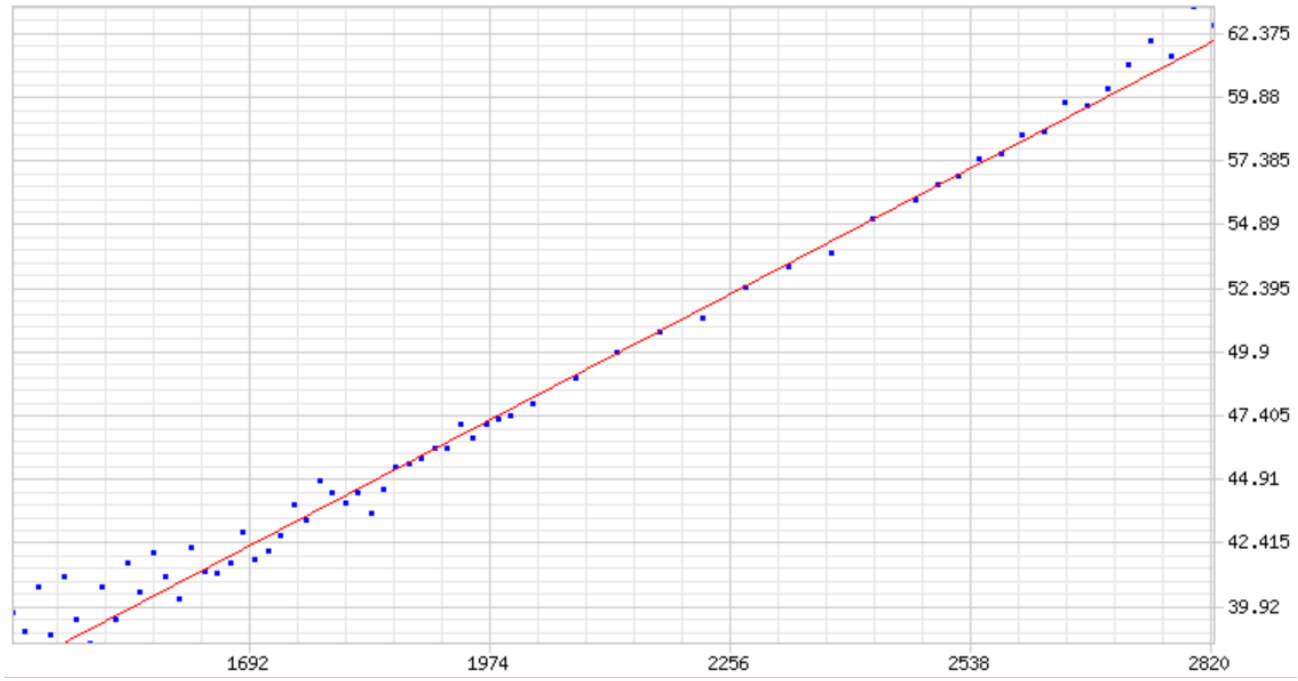
T1 match: $R^2 = 0.991$



With 300-point compression---less good at $R^2 = 0.986...$:

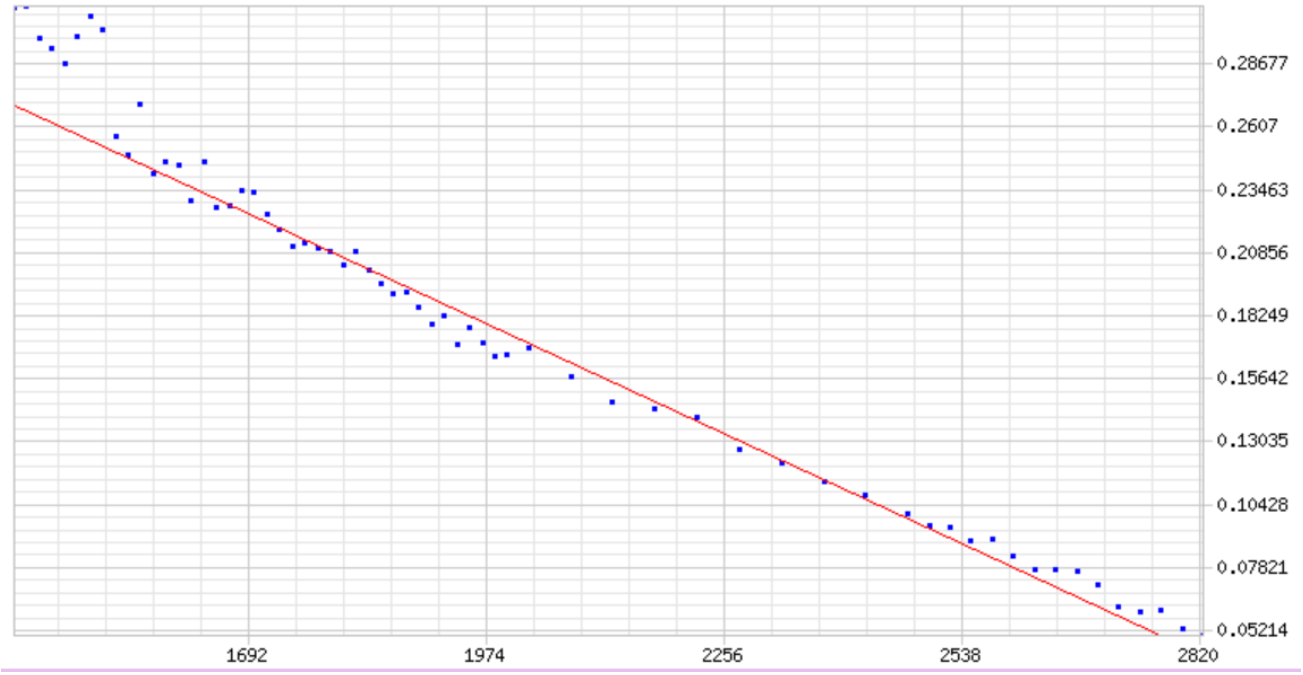


EV: $R^2 = 0.988...$



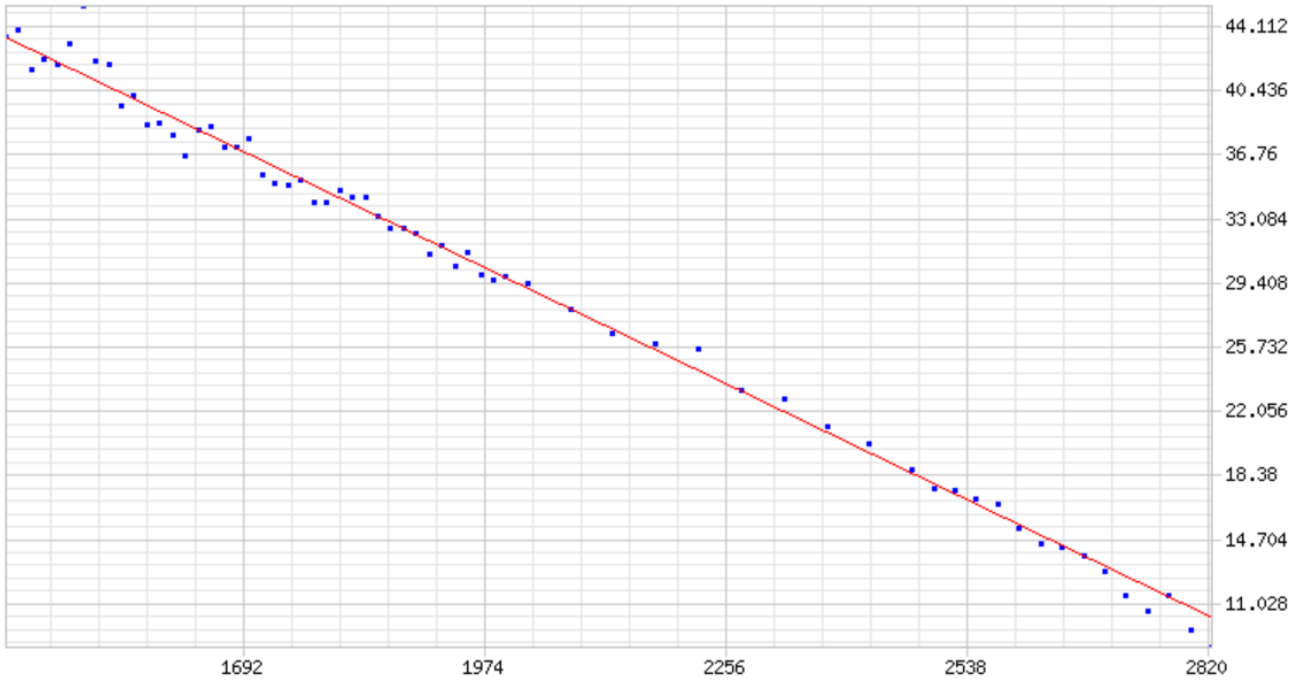
The Sonas 300, however, is more noticeably worse, with R^2 "only" 0.9789...

ASD: $R^2 = 0.9575...$

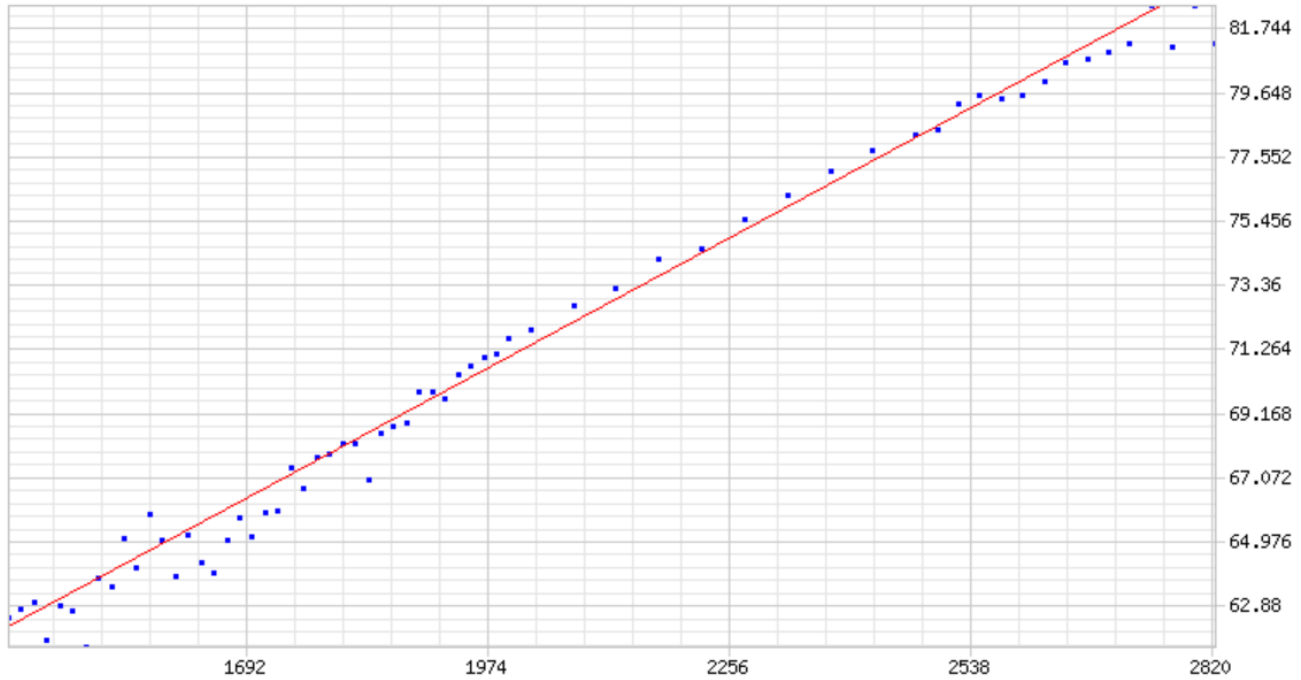


The Sonas 300 hits $R^2 = 0.971...$, still not as good. (But see ASD runs at the very end.)

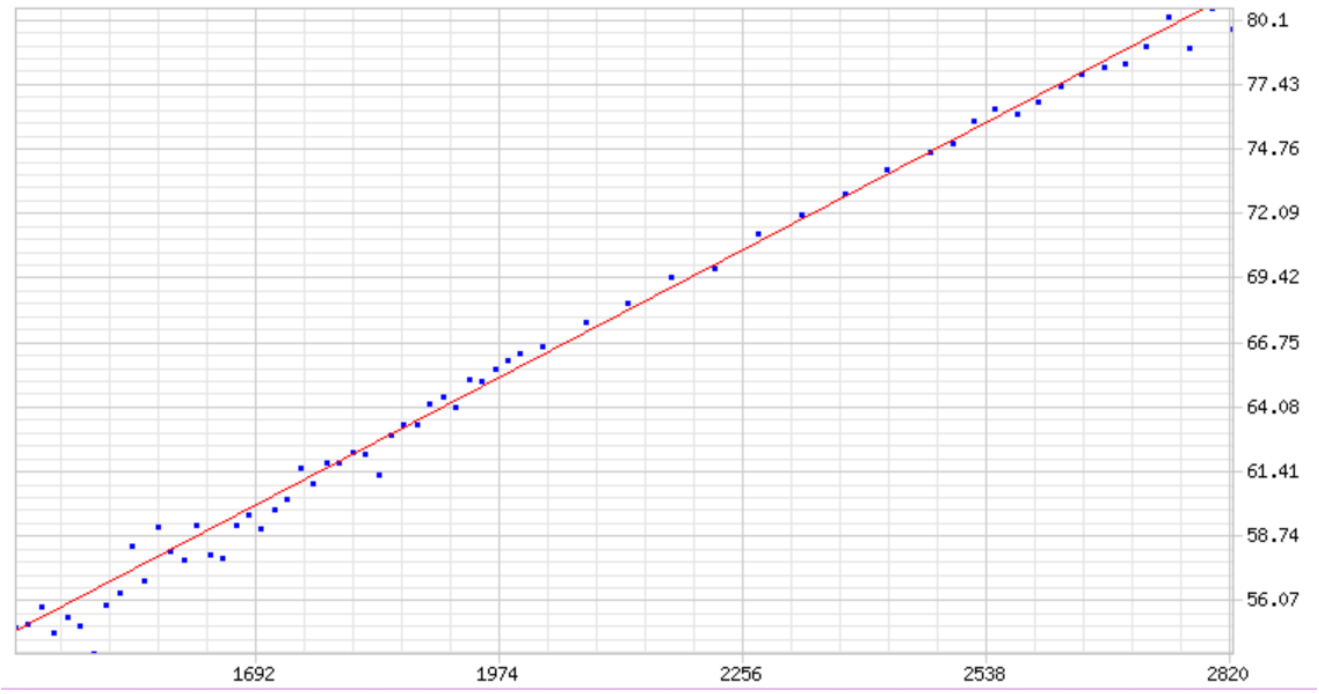
Error 0.25 Count: superb---$R^2 = 0.9928...$ (Sonas 300 slightly worse at 0.9905...)



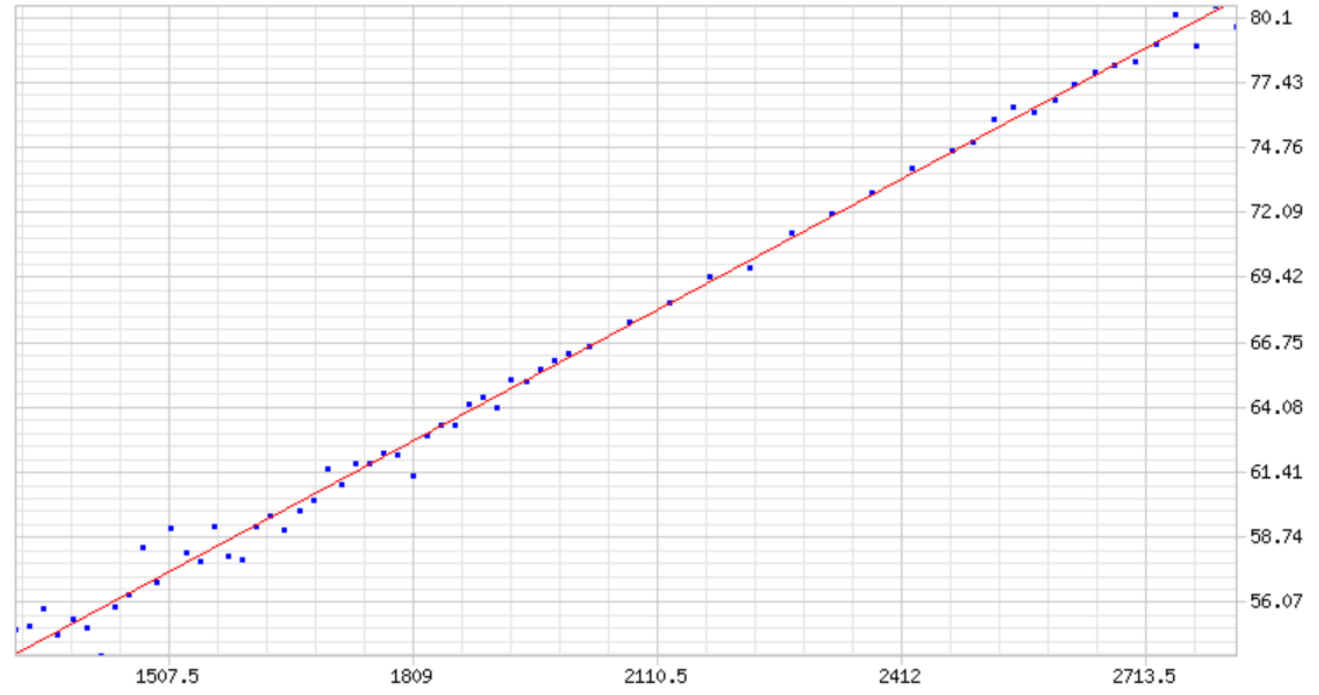
T3 Match: $R^2 = 0.9863...$



T3 Match with 0.50 error cutoff is even better after the fix: $R^2 = 0.9926...$:

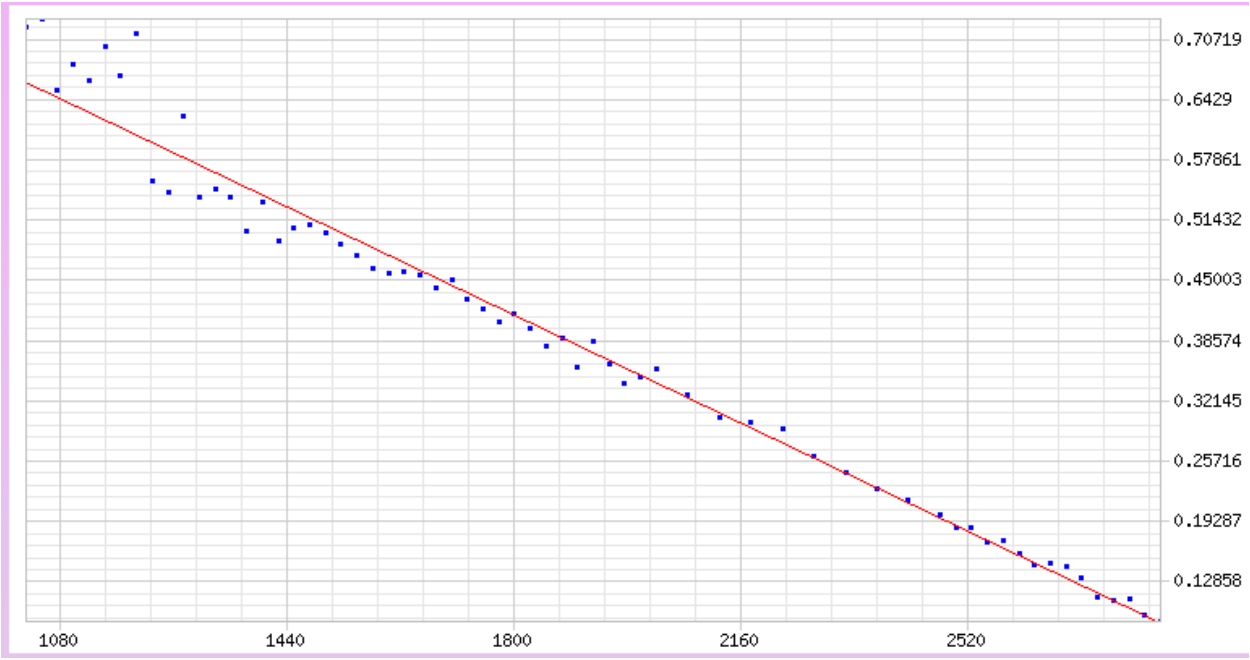


And actually, the Sonas 300 is better here: $R^2 = 0.9944...$

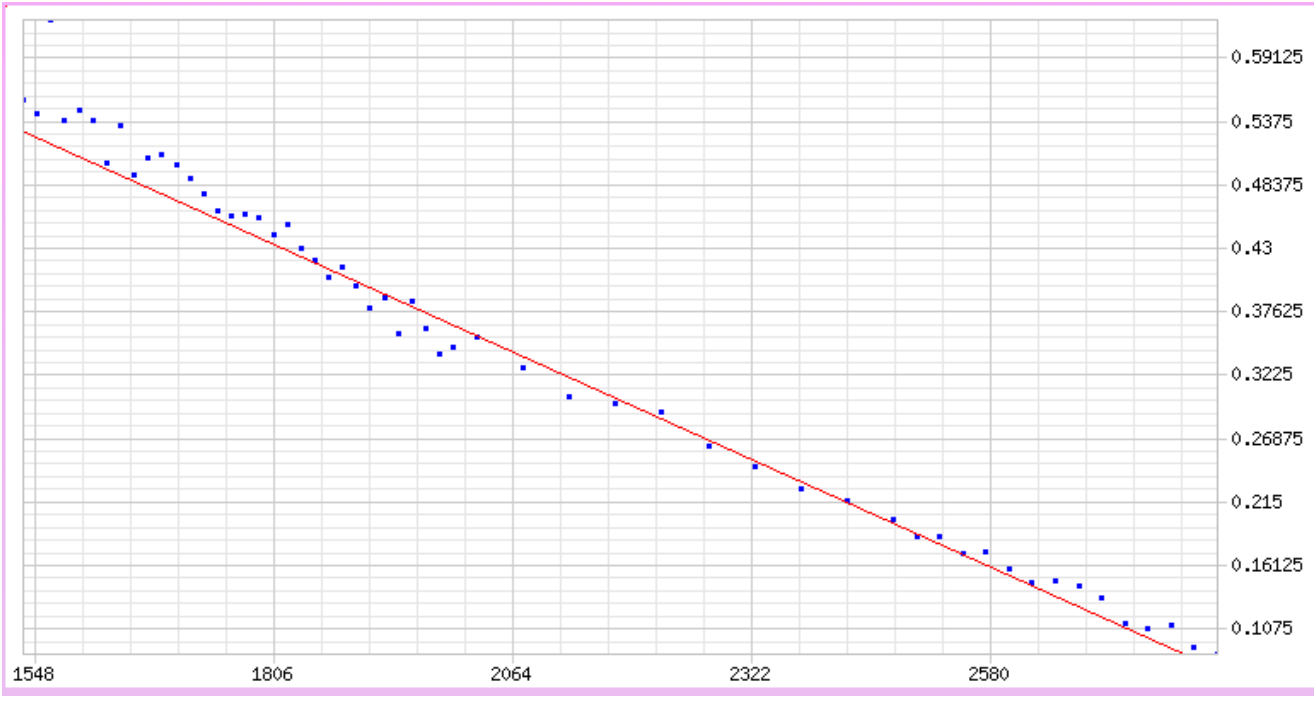


Using data from 2016--2019 only: Sonas 300 gives $R^2 = 0.991...$, while Sonas 400 gives $R^2 = 0.987...$ This is only faint support for the idea that the rating skew got progressively worse thru 2019. Before the pandemic I might have tried to argue the fix down to 300, but absent a reliable way to test things over 2020--2023, I support the full 400-point fix.

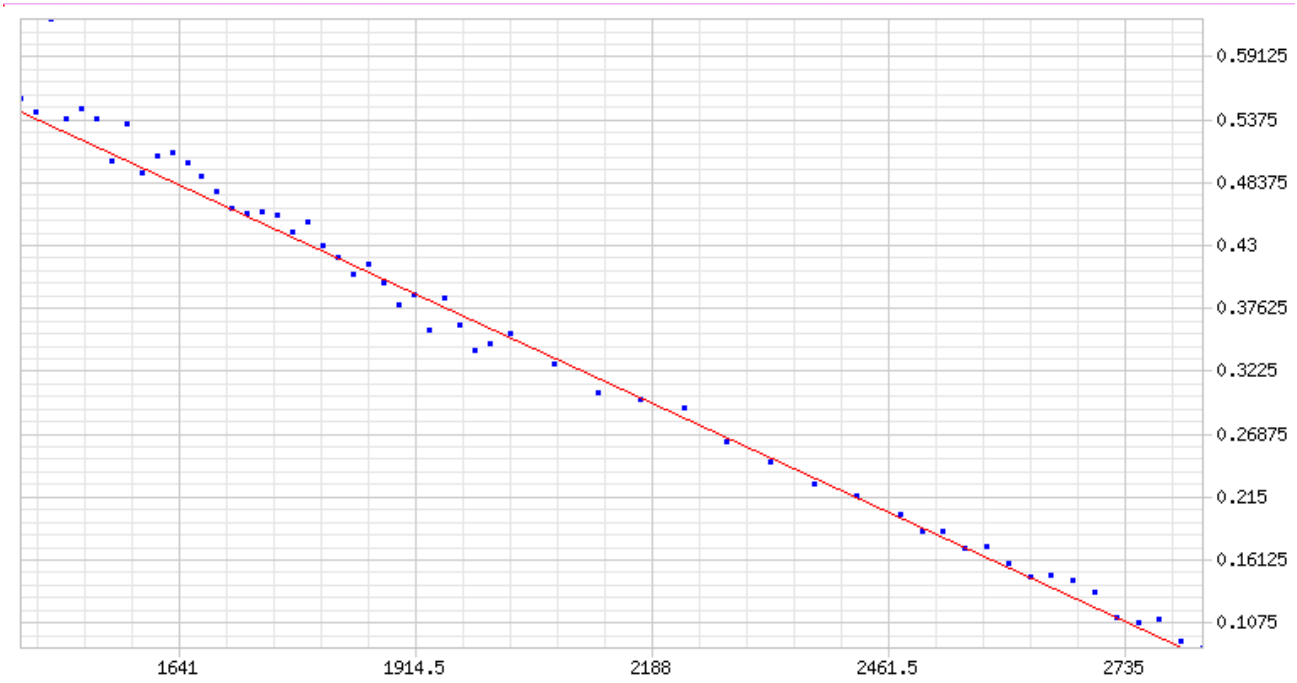
Unscaled ACPL---a nice line is good all the way down to 1200. $R^2 = 0.9764...$



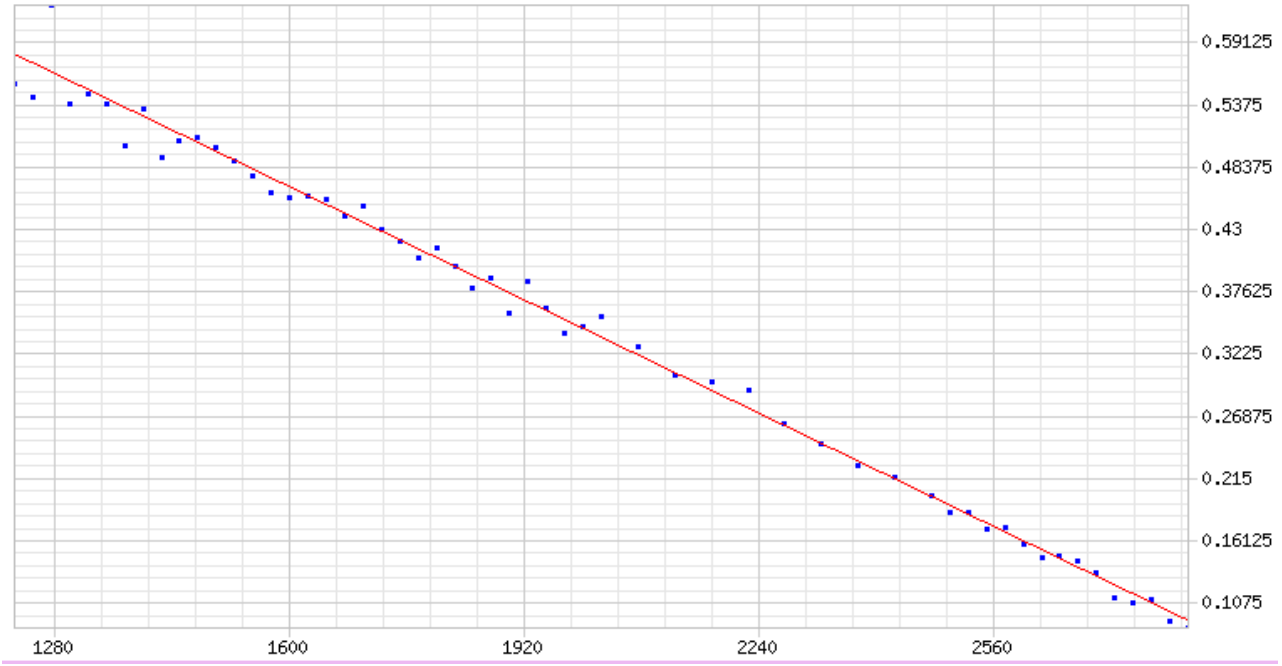
After 400 compression: bad ($R^2 = 0.9371...$) but only $\leq$ 1200. Pruning the bottom 8 data points:



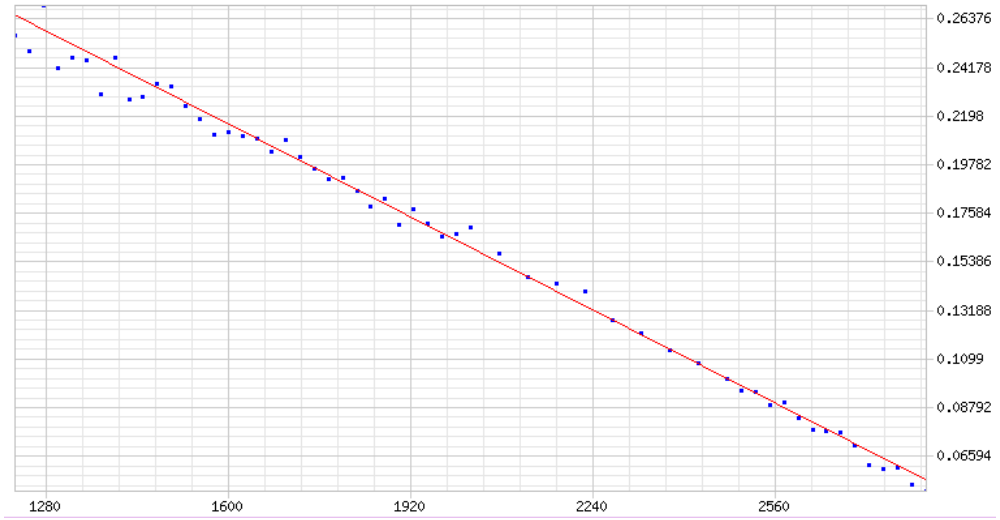
This gives $R^2 = 0.9789...$ Using the 300 compression instead, fully fine at $R^2 = 0.987...$:



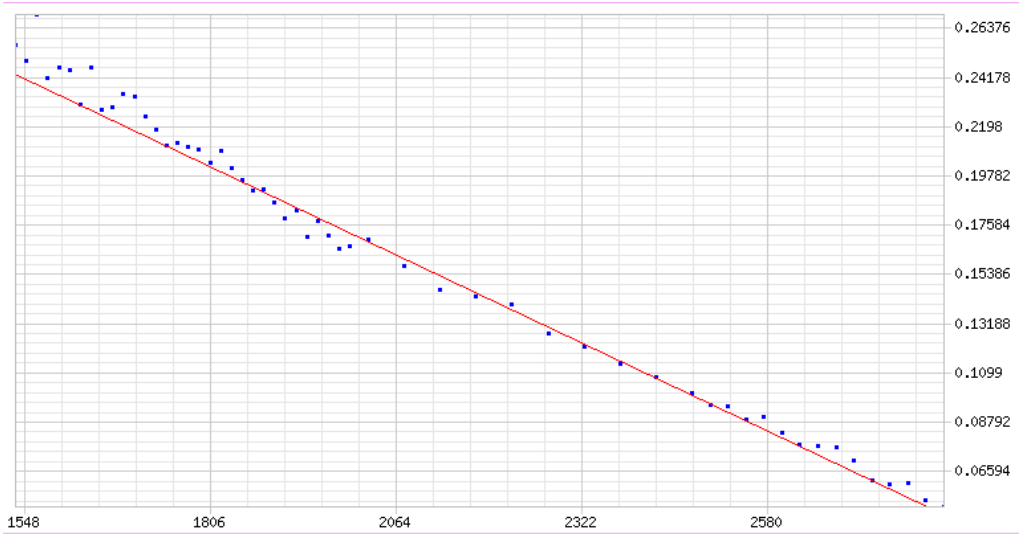
Full cycle back to no compression with this pruning:



Well, this is still better with $R^2 = 0.9919...$ The main takeaway, however, is that one can still argue a linear relation with ACPL (and ASD) after the 400 compression. With ASD after the same pruning ($R^2 = 0.9924...$):



400 compression, still a little "overcooked", $R^2 = 0.9859...$



With 300 compression: just as good at $R^2 = 0.9922...$: